Algorithm Driven Design

Comparison of Single-Objective and Multi-Objective Genetic Algorithms in the Context of Housing Design

Serkan Kocabay ¹, Sema Alaçam ¹

¹ Istanbul Technical University

architectserkan@gmail.com, semosphere@gmail.com

Abstract. This paper aims to present a dynamic multi objective genetic algorithm (MOGA) framework for the purpose of generating 3D mass models in the context of housing design. The proposed MOGA framework contains static and dynamic modules such as regulations, environmental condition analysis as static, behavioral models, designer-specified goals, domain-specific goals based on building types as dynamic modules. Moreover comparison of two algorithmic approaches, implementation of a single and multiple objective genetic algorithms are compared in terms of variety and usability of the generated design solutions, fitness approximation performances and the speed of the algorithms (running time). In the scope of this study, the potentials and limitations of the proposed MOGA framework in 3D form generation, its advantages over single objective genetic algorithm are discussed, conducted with a case study.

Keywords: Multi-objective, Genetic Algorithm, Housing Design, Mass-model

1 Genetic Algorithms in Architectural Design

Studies in computer science on natural evolution metaphor have led to the emergence of well-established algorithms. As a result a diverse range of methodologies, techniques, and concepts have emerged such as genetic algorithms, evolutionary algorithms and artificial ways of mimicking natural selection. The noticeable increase in the use of algorithms in architectural design process also overlaps with the Carpo’s ‘Digital Turn’ conception [1] which indicates the transformation in the way of thinking and making in architecture since early 1990s. Genetic algorithms (GAs) which was mainly developed for machine learning and optimization problems [2, 3, 4, 5] further bred various methodologies and new ways of thinking for different contexts. Although the theoretical contributions can be traced back to 1970s [6, 7] and there have been discussions on the potentials of GAs in conceptual design process since 1990s [8, 9], earlier implementations of GA in architectural design have been focusing on well-defined design problems and optimization processes. In relation with
the reflection process of GAs onto architectural design, three considerations are listed below:

Representation problem: How design knowledge is represented and translated into computer affects the whole process and the design outcomes as well. In the very beginning of the algorithm development, flexibility of the assumptions, variables and fitness function matters. The more flexible fitness function is defined, the more population among design alternatives might be selected for the further steps.

Rigidity of the method and concepts: Well-defined algorithmic structures and their related concepts such as genotype, phenotype, fitness function, crossover, selection, mutation, etc. have been inherited from computer science. New visionary contributions in which architects play a pioneering role in the conceptualization process are needed.

Domain-specific or objective-specific nature of GAs: There might be difficulties in approaching the well-defined and ill-defined design problems simultaneously. Here, the terms well-defined and ill-defined refer to Hayes’ and Ormerod’s definitions for problem-solving strategies [10, 11]. Non-dominated sorting algorithms have a potential to partially respond to providing a shared ground for evaluation of different goals simultaneously.

This study aims to contribute to developing better understanding and awareness in regard to the third problem: multi-objectivity in GAs and simultaneous execution of multiple inputs. The wide-ranging adaptation of GAs in architectural design is mostly focus on hybrid combination of two initial genotypes based on definition of a fitness function. The flexible fitness function is defined, the more population among design alternatives might be selected for the further steps. If multiple fitness functions are connected to each other in a serial way, each time there will be reduction based on this singular penalties which will cause elitism. However, regarding the nature of architectural design process, more iterative and recursive decision making environments are needed. Multi-objective genetic algorithmic dynamic modules have potential to response the required flexibility. User interfaces giving opportunity to add new customized modules are crucial in the case of multi-objective genetic algorithms (MOGA), apart from the simultaneous connection of different design goals. This study presents an integrated multi-objective genetic algorithm framework proposal, its implementation in a housing design project in comparison with the outcomes achieved from single-objective genetic algorithms.

2 Towards Non-Routine Design: Evolutionary MOGA

The concept of multi-objective genetic algorithm (MOGA) was first introduced by Schaffer, in his paper entitled “multi objective optimization with vector evaluated genetic algorithms”. Schaffer [12] contributed towards engagement of multi optimization objective problems and genetic algorithms. Following this research several multi objective evolutionary algorithms [13, 4] have been studied under different topics and titles. “Multi-objective Genetic Algorithm” (MOGA) was introduced by [14], “Niched Pareto 6 Genetic Algorithm” was introduced by [15] “Random Weighted Genetic Algorithm” (RWGA) by [16] and “Nondominated Sorting Genetic Algorithm” (NSGA) by [17]. In the second half of 1980s and 1990s, while there have been a considerable progress in the specification of the multi
objective approaches, these methods were mostly developed and used by engineers in defined problems of sorting and optimization. New contributions continued in the following decade such as “Strength Pareto Evolutionary Algorithm” (SPEA) [18], “Pareto-Archived Evolution Strategy” (PAES) [19], Fast Non-dominated Sorting Genetic Algorithm (NSGA-II) [20], Multi-objective Evolutionary Algorithm (MEA) [21], Rank-Density Based Genetic Algorithm (RDGA) [22]. Introducing the “Adaptive Weight Sum Method”, Kim and Weck [23] pointed out the future directions of the multi-objective genetic algorithms: “We propose a new adaptive method, based on the weighted-sum approach, for multiobjective optimization. In this approach, the weights are not predetermined, but they evolve according to the nature of the Pareto front of the problem” [23, p.150]

Briefly, genetic algorithms (GAs) are capable of responding to the problems which involve multiple objectives. However, As Rosenman [24] stated, until mid 2000s GAs had been used for optimization and machine learning problems with a few exceptions. To mention, Frazer’s [8] theoretical contributions or Elezkurtaj and Frank’s [25]’s explorations in the implementation of artificial evolutionary approaches in architectural floor plan design might be considered as promising studies from 1990s. As one of the earlier theoretical contributors, [24] discussions in terms of adaptation of GAs into non-routine design process have opened new directions.

The accumulation of experience not only in application of GAs in design but also usage of various computational approaches in design resulted with a significant paradigm shift in 2000s. Therefore beyond the consideration of GAs merely a sorting and optimization method, new approaches and interpretations were emerged in which GAs were became active agents of integrated design approaches. Instead of being used after most of the design decisions are taken, GAs and later MOGAs reflected onto the conceptual design processes [26, 27]. For instance, [26] “a forest of columns” assumption led an expansion in the meaning of GAs towards a metaphorical interpretation. Scheurer’s [26] approaching GAs in form-finding process and structural optimization also affected the definition of initial parameters and earlier phases of design process. Since 2000s, the tension between non-routine nature of design and routine characteristic of the engineering method GAs resulted with various novel approaches involving multiple and nondominated objective strategies. On one hand, finding a set of non-dominated solutions among multiple objectives can be still considered as a routine process in which all the parameters are expected to be defined from the beginning, the boundaries of the solution space and/or number of the outcomes are finite. On the other hand, simultaneous search for different parts of the solution space have potential to respond to complex problems via creating a finite number of but diverse set of solutions.

3 A Framework Proposal for MOGA in Housing Design Design

A framework proposal for multiobjective genetic algorithm (MOGA) is introduced in this section. The proposed framework (Fig. 1) consists of modular components, therefore it is possible to add, remove or update any module if required. The implementation of the framework was developed in Rhino Grasshopper environment conducted with scripting and add-ons including Rhino/Octopus and Rhino/Galapagos. Figure 1 shows the selected objectives and their sub-components such as regulations,
physical environmental parameters as static, behavioral models, designer-specified goals, domain-specific goals based on building types as dynamic modules. As it is explained in Section 4 in detail, the gray sub-components (Fig. 1) are defined and used in the implementation model.

<table>
<thead>
<tr>
<th>Modular Structure</th>
<th>Selected Items</th>
<th>Inputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regulations</td>
<td>Zoning Regulations</td>
<td>-Distance between building limits</td>
</tr>
<tr>
<td></td>
<td>Fire Regulations</td>
<td>-Obstacles from the peripheral area</td>
</tr>
<tr>
<td></td>
<td>Earthquake Regulations</td>
<td>-The balance of area usage regulates the ratio of floor area and maximum height</td>
</tr>
<tr>
<td>Physical Environmental Parameters</td>
<td>Solar Analysis</td>
<td>-Solar Radiation</td>
</tr>
<tr>
<td></td>
<td>Wind Analysis</td>
<td>-Wind load</td>
</tr>
<tr>
<td></td>
<td>Aquatic Analysis</td>
<td>-Exodermal resistance</td>
</tr>
<tr>
<td>Domain Specific Design Goals</td>
<td>Residential</td>
<td>-Exodermal relation</td>
</tr>
<tr>
<td></td>
<td>Commercial</td>
<td>-Ground usage efficiency</td>
</tr>
<tr>
<td></td>
<td>Leisure</td>
<td>-Topological control</td>
</tr>
<tr>
<td></td>
<td>Mixed Use</td>
<td>-Percentage of buildings</td>
</tr>
<tr>
<td>Designer Specified Goals</td>
<td>G1</td>
<td>-Ground floor efficient area usage control</td>
</tr>
<tr>
<td></td>
<td>G2</td>
<td>-Floor -read control</td>
</tr>
<tr>
<td></td>
<td>G3</td>
<td>-Morphological shell control</td>
</tr>
<tr>
<td>Urban/Global Analysis</td>
<td>Transportation</td>
<td>-Transportation zones effects ground floor usage</td>
</tr>
<tr>
<td></td>
<td>Accessibility</td>
<td>-Environmental public accessiblity control</td>
</tr>
<tr>
<td></td>
<td>City data</td>
<td>-Environmental contextual data effects</td>
</tr>
</tbody>
</table>

Fig. 1. Algorithm schema for the proposed framework

The regulation module is related to the context-dependent design requirements. The number of the sub-components and definitions might be different in each architectural design process. In the scope of this study, zoning regulations of the selected site for housing design was used. Designer Specified Goals module can be considered as one of the most flexible module which can be developed, adapted or changed according to subjective design criteria. The selection among the design solutions according to the fitness functions are made simultaneously. However, the modules have different impact factors affecting the results. Moreover these impact factors can be changed by the users interactively. In the implementation process the MOGA framework was firstly defined by using multi objectives in Rhino/Grasshopper environment and Octopus add-on. In order to compare the performance of multi-objective and singular-objective approaches, later the algorithm was converted to single objective genetic algorithm (SOGA) (Fig. 2).
In the conversion process multiple fitness functions are connected to each other in a serial way, each time, there is a level of reduction based on these singular penalties. Adapting the algorithms to Grasshopper components (both in Octopus and Galapagos), each objective's weight was defined in a range. In this step, in order to convert MOGA to SOGA, [28] weighted sum model was used (Fig. 3).

\[
\text{When } x = f(x), y = \text{def. above} \rightarrow \begin{cases} 
\text{IF} \left( \text{FLOOR AREA RATIO CONTROL} \geq \text{TRUE ; } x, y \right) \times 10 + \text{IF} \left( \text{LOT COVERAGE RATIO} \leq \text{TRUE ; } x, y \right) \times 0.5 \\
\text{IF} \left( \text{PEDESTRIAN CONTROL} \leq \text{TRUE ; } x, y \right) \times 0.5 + \text{IF} \left( \text{EFFICIENCY CONTROL} \leq 1.7 ; x, y \right) \times \text{RM}^* \left( x \leq 1.0 \right) + \text{IF} \left( \text{FACADE CONTROL} \leq 0.8 ; x, y \right) \times \text{RM}^* \left( x \leq 1.0 \right) \\
\end{cases}
\]

Fig. 3. Implication of Weighted Sum Model

4 Implementation of Single and Multi Objective Generative Algorithms in Housing Design

The case study involves implementation of the proposed framework for MOGA in Rhino Grasshopper-Octopus and SOGA in Grasshopper-Galapagos. The MOGA and SOGA models are used for the purpose of generating 3D model alternatives at a defined site, Fikirtepe. Variety and usability of the generated design solutions, fitness approximation performances and the speed of the algorithms (running time) of the models are compared.

4.1 Selected Site and Problem Definition

Fikirtepe was chosen as a case study area in this project. Fikirtepe, as a settlement area consisting mostly of one storey residence, is located in Istanbul-Kadıköy in Turkey. Fikirtepe was declared as an urban transformation zone by the Law on the Transformation of Areas under Disaster Risk (Law No.6306, 31.05.2012). As it is in the urban transformation zone, the floor area ratio is fixed at 4.00 which was to encourage contractors, and rapidly changing urban identity is making design conditions difficult. Rapidly changing urban identity shortens the deadlines of designs and the high floor area rate limits morphological diversity. For these reasons,
Fikirtepe was determined to benefit from the computational methods in the early phase of the design process for generating housing design proposal. Using computational design methods brings not only time optimization and variety, but also a rich pool of solutions, including well-defined natural parameters that are not normally well defined but intuitively used by designers.

Fig. 4. Octopus (MOGA) vs Galapagos (SOGA) algorithm process diagram

In the conversion process multiple fitness functions are connected to each other in a serial way, each time, there is a level of reduction based on these singular penalties. Adapting the algorithms to Grasshopper components (both in Octopus and Galapagos), each objective’s weight was defined in a range. In this step, in order to convert MOGA to SOGA, [28] weighted sum model was used (Fig. 3).

Fig. 3. Implication of Weighted Sum Model

4. Implementation of Single and Multi Objective Generative Algorithms in Housing Design

The case study involves implementation of the proposed framework for MOGA in Rhino Grasshopper - Octopus and SOGA in Grashopper - Galapagos. The MOGA and SOGA models are used for the purpose of generating 3D model alternatives at a defined site, Fikirtepe. Variety and usability of the generated design solutions, fitness approximation performances and the speed of the algorithms (running time) of the models are compared.

4.1 Selected Site and Problem Definition

Fikirtepe was chosen as a case study area in this project. Fikirtepe, as a settlement area consisting mostly of one storey residence, is located in Istanbul - Kadıköy in Turkey. Fikirtepe was declared as an urban transformation zone by the Law on the Transformation of Areas under Disaster Risk (Law No. 6306, 31.05.2012). As it is in the urban transformation zone, the floor area ratio is fixed at 4.00 which was to encourage contractors, and rapidly changing urban identity is making design conditions difficult. Rapidly changing urban identity shortens the deadlines of designs and the high floor area rate limits morphological diversity. For these reasons, Fikirtepe was determined to benefit from the computational methods in the early phase of the design process for generating housing design proposal. Using computational design methods brings not only time optimization and variety, but also a rich pool of solutions, including well-defined natural parameters that are not normally well defined but intuitively used by designers.

Fig. 4a. Fikirtepe Location in Istanbul 4b. Design site location in Fikirtepe

Fig. 4c. Design Site and Neighbourhood Relations 4d. Design Site Pedestrian Connections

In the beginning of the design process, the 13133 m² floor area and the maximum construction volume of 33739 m³, 4700 m² as cession of territory for municipality, 80m as a maximum height and the regulation values were accepted as predefined criteria.

4.2 Parameterization of the Design Objectives

In the scope of the case study, 5 modules and 8 sub-components are used as individual design goals with their own fitness functions. The distances between the building blocks (b), the distances to the boundary of the constructible area (a), the floor area ratio of the site (FAR) and the maximum height (hmax) are selected as regulation parameters. An additional objective is defined by the authors for checking the effectiveness of site usage at the masterplan level. As an environmental control module, an existing solar analysis tool is integrated with the algorithm. Another module used in case study is an agent-based pedestrian movement simulation. Connections between the selected area and the existing street nodes are used as a basis for the creation of pedestrian movement simulation. In terms of domain specific
design goals, housing units are selected and defined as a node in a matrix-based layout. Pedestrian movement simulations between different objectives were carried out only once at the beginning of the evaluation process due to heavy load. The objectives are checked in the tolerance range to maintain beneficial results. These criteria of the design objectives and sub-components are formulated below:

![Site notations](image)

**Fig. 5.** Site notations

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SA</td>
<td>Site area</td>
</tr>
<tr>
<td>SB</td>
<td>Site boundary</td>
</tr>
<tr>
<td>FAR</td>
<td>Floor area ratio</td>
</tr>
<tr>
<td>CAV</td>
<td>Constructable area volume</td>
</tr>
<tr>
<td>LCR</td>
<td>Lot coverage ratio</td>
</tr>
<tr>
<td>CoT</td>
<td>Cession of territory for municipality</td>
</tr>
<tr>
<td>hfl</td>
<td>Floor height</td>
</tr>
<tr>
<td>hhs</td>
<td>High structure limit</td>
</tr>
<tr>
<td>hmax</td>
<td>Maximum height of constructed building</td>
</tr>
<tr>
<td>SM</td>
<td>Solar Mass</td>
</tr>
<tr>
<td>Eb</td>
<td>Longest edge of the outer bounds of the design site</td>
</tr>
<tr>
<td>Ec</td>
<td>Longest edge of the outer bounds of the current modules</td>
</tr>
<tr>
<td>Fb</td>
<td>Modularization of outer bounds of the facade silhouette</td>
</tr>
<tr>
<td>Fc</td>
<td>Modularization of current bounds of facade silhouette</td>
</tr>
<tr>
<td>a</td>
<td>Constructable area distance</td>
</tr>
<tr>
<td>b</td>
<td>Distances between high rise attraction points of building blocks</td>
</tr>
<tr>
<td>c</td>
<td>Module axis measurement</td>
</tr>
<tr>
<td>n1</td>
<td>Number of total modules</td>
</tr>
<tr>
<td>n2</td>
<td>Number of floor modules</td>
</tr>
<tr>
<td>xm</td>
<td>Rising points coordinates</td>
</tr>
</tbody>
</table>

**4.2.1 Regulation Module**

Regulation criteria are coded based on the constraint defined by local municipalities (Fig. 6a) in zoning regulations sub-module. A boundary shape was created to describe a volumetric constraint to represent the constructible area (Fig. 6b). This zoning regulation sub-module checks whether the generated floor area is in the tolerated range (Fig. 6c). The number interval is calculated based on the given zoning regulation. The total constructible area was checked with an additional tolerance constant to adapt it to Galapagos interface (Fig. 6d).
design goals, housing units are selected and defined as a node in a matrix-based layout. Pedestrian movement simulations between different objectives were carried out only once at the beginning of the evaluation process due to heavy load. The objectives are checked in the tolerance range to maintain beneficial results. These criteria of the design objectives and sub-components are formulated below:

**Fig. 5. Site notations**

- **SA** = Site area
- **SB** = Site boundary
- **FAR** = Floor area ratio
- **CAV** = Constructable area volume
- **LCR** = Lot coverage ratio
- **CoT** = Cession of territory for municipality
- **hfl** = Floor height
- **hhs** = High structure limit
- **hmax** = Maximum height of constructed building
- **SM** = Solar Mass
- **Eb** = Longest edge of the outer bounds of the design site
- **Ec** = Longest edge of the outer bounds of the current modules
- **Fb** = Modularization of outer bounds of the facade silhouette
- **Fc** = Modularization of current bounds of facade silhouette
- **a** = Constructable area distance
- **b** = Distances between high rise attraction points of building blocks
- **c** = Module axis measurement
- **n1** = Number of total modules
- **n2** = Number of floor modules
- **xm** = Rising points coordinates

### 4.2.1 Regulation Module

Regulation criteria are coded based on the constraint defined by local municipalities (Fig. 6a) in zoning regulations sub-module. A boundary shape was created to describe a volumetric constraint to represent the constructible area (Fig. 6b). This zoning regulation sub-module checks whether the generated floor area is in the tolerated range (Fig. 6c). The number interval is calculated based on the given zoning regulation. The total constructible area was checked with an additional tolerance constant to adapt it to Galapagos interface (Fig. 6d).

**Fig. 6.**

- **Fig. 6a. Regulation criteria**
- **Fig. 6b. Constructable volume**
- **Fig. 6c. Constructable floor area**
- **Fig. 6d. Constructable area**

The formulas used in the zoning regulation sub-module’s algorithm are shown below:

To maximize the constructible area

\[ (SA-CoT)*FAR/(c^2*n1) < 3000 \]  

(1)

To check the lot coverage check

\[(SA*IC)-n2>0\]  

(2)

To create the constructible area volume

Loft (Move (Offset SB curve x=distance when(x= 0 to [integer(hhs/hfl)] and y=5 If {x≤4, y,[((x-4)*0.5+y)}))) in Z direction x = distance when (x=0 to Z direction x=distance when (x=integer[(hmax-hhs)/hfl]]) 15 + (x*0.5) in Z direction x=distance when (x=hhs to hhs +{3* integer[(hmax-hhs)/hfl]})

(3)

### 4.2.2 Physical Environmental Parameters

The solar performance of the outputs are restricted by a dynamic solar control mechanism based on the location information of the site (Fig.7).

**Fig. 7.**

- **Fig. 7. Checking the solar efficiency ratio [28]**

Maximizing the solar efficiency

\[ -\sum SM-MIN \]  

(4)
4.2.3 Domain Specific Design Goals

A gridal axis system is constructed to represent to modules of housing units. The grid nodes are multiplied in the Z direction to create a 3dimesional axis system. (Figure 8a) Rising attraction points as a dynamic gene-pool is constructed (Fig. 8b)

Multiplication of modules in Z direction x=distance when
\[ x=0 \text{ to } (\text{Intersection in Z direction } x_m \text{ rays with CAV}) \]

\[ \text{(5)} \]

4.2.4 Designer Specified Goals

\[
\begin{align*}
\text{FAÇADE CHECK} & : & \frac{\text{USED FAÇADE MODULE COUNT}}{\text{FULL FAÇADE MODULE COUNT}} & < 0.8 \\
\text{EDGE CONTROL} & : & \frac{\text{SITE LONGEST EDGE}}{\text{CONSTRUCTED AREA LONGEST EDGE}} & < 1.7
\end{align*}
\]

\[ \text{Fig. 9. The density checker objective \ [29]} \]
A facade density system was developed to calculate the percentage of full-to-empty areas of products. The use of the construction site was limited to a fixed value to avoid inefficient solutions (Figure 9). 
\[
\text{Fb/Fc} < 0.8 \text{ and } (\text{Fb/Fc}) \rightarrow \text{MIN} \quad \text{(Figure 9).} 
\]

The edge checking system is generated to prevent inefficient usage of the site 
\[
\text{Eb/Ec} < 1.7 \text{ and } (\text{Eb/Ec}) \rightarrow \text{MIN} 
\]

4.2.4 Designer Specified Goals

Fig. 9. The density checker objective [29]

A facade density system was developed to calculate the percentage of full-to-empty areas of products. The use of the construction site was limited to a fixed value to avoid inefficient solutions (Figure 9).

\[
\text{Fb/Fc} < 0.8 \text{ and } (\text{Fb/Fc}) \rightarrow \text{MIN} \quad \text{(Figure 9).} 
\]

The edge checking system is generated to prevent inefficient usage of the site

\[
\text{Eb/Ec} < 1.7 \text{ and } (\text{Eb/Ec}) \rightarrow \text{MIN} 
\]

4.2.5 Urban-Global Analysis

Fig. 10. Urban/Global Analysis [29]

Location and topographic information are derived from the rhino interface. The borders of the site are defined in the Rhino interface. Pedestrian analysis was created using environmental parameters. The circulation area was maximized. Applying site restrictions to borders (according to the law) to create a constructive volume of the site. The final result was gathered from pareto frontal solutions (Fig.10).

5 Finding, Outcomes and Evaluation

Based on the design objectives explained in Section 4, both MOGA (Octopus) and SOGA (Galapagos) were set to run 10 hours same settings. The findings and outcomes are discussed below.
5.1 MOGA (Octopus) SOGA (Galapagos) Outcomes

Selected outcomes derived from MOGA are shown below (Fig. 11). Number of the generations are shown on the left, 3D mass models shown in the other columns. Pareto optimal non dominated solutions are represented as final result of each generation. A wide range of variations are derived from SOGA compared to MOGA.

![Fig. 11. Final outputs of Octopus generations [29]](image)

SOGA(Galapagos) outcomes were sorted by generations shown below (Fig. 12). For each generation best of four solutions were represented in the diagram on the left column of Fig. 12. The solutions get closer to the fitness function by early 92 generations. At this generation the outputs matched with the fitness function very well and the morphological similarities between them were also high.
Selected outcomes derived from MOGA are shown below (Fig. 11). Number of the generations are shown on the left, 3D mass models shown in the other columns. Pareto optimal non-dominated solutions are represented as final result of each generation. A wide range of variations are derived from SOGA compared to MOGA.

Fig. 11. Final outputs of Octopus generations [29]

SOGA (Galapagos) outcomes were sorted by generations shown below (Fig. 12). For each generation best of four solutions were represented in the diagram on the left column of Fig. 12. The solutions get closer to the fitness function by early 92 generations. At this generation the outputs matched with the fitness function very well and the morphological similarities between them were also high.

Fig. 12. Galapagos generations 1 to 92

As it is seen Fig. 12, in SOGA the system gained a kind of equilibrium state after the 92th generation, on the other hand MOGA (Fig. 13) still provides a degree of variety for the design solutions in the higher generations. The more the components of SOGA gets closer to the fitness function, the earlier formal explorations are finalized. In the scope of this stud, from the point of design or user, dealing with a diverse range of outcomes can be accepted as an advance of MOGA.
When the generations between 100th and 300th are examined, it was observed that the outputs are getting far from the fitness function by effect of mutations and after a while it stays stabilized. After 10 hours of calculations Galapagos has stop process at 312th generation.
The diagrams above shows the process of Octopus and Galapagos. They both have meet the fitness function needs. When they examined it can be say that Octopus keep finding a pareto front mesh which meets the needs of fitness function at any generations, but Galapagos reaches and loses the fitness function and got stabilized after a while.

5.2 Comparison of MOGA and SOGA in the Context of Morphological Relations

The evaluation of the outcomes of MOGA and SOGA is shown in Table 1 below:

<table>
<thead>
<tr>
<th>MOGA (Octopus)</th>
<th>SOGA (Galapagos)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Using SPEA2 MO Algorithm</td>
<td>Using Basic Genetic Algorithm</td>
</tr>
<tr>
<td>The solution space is 3D</td>
<td>The solution space is 2D</td>
</tr>
<tr>
<td>Outputs has a wide range of variants</td>
<td>Outputs has a limited range of variants</td>
</tr>
<tr>
<td>More generations needs to reach the fitness function</td>
<td>Less generations needs to reach the fitness function</td>
</tr>
<tr>
<td>A non dominated pareto front pool is generated</td>
<td>A dominated objective pool is generated</td>
</tr>
<tr>
<td>Has default tools to define objectives and penalties</td>
<td>System needs to be optimized to define objectives and penalties</td>
</tr>
<tr>
<td>Many outputs can be baked while using the program</td>
<td>Only 1 outcome can be bake using the program</td>
</tr>
<tr>
<td>Let user see different density of objectives</td>
<td>Focuses defined objectives and limits the potentials of variables</td>
</tr>
<tr>
<td>simultaneously</td>
<td></td>
</tr>
</tbody>
</table>

Comparing two algorithms, some differences are gathered which is shown above. These results show that using MOGA and SOGA is relatively affects the design process, in the case studies it was seen that more alternatives are gathered in the MOGA system, which are evaluable in the generation process. In other hand, SOGA system is faster at one single problem solving but the generation results are not evaluable during process. Also MOGA system try to get better results using a pareto front mesh and allow user see possible results, while SOGA system only use a section of that mesh.

6 Conclusion and Discussion

This paper presents a framework proposal for a multi objective genetic algorithm
(MOGA) and its comparison with a single objective genetic algorithm (SOGA) in a case study. The proposed MOGA and SOGA framework contains static and dynamic modules such as regulations, environmental condition analysis as static, behavioral models, designer-specified goals, domain-specific goals based on building types as dynamic modules. The modules are selected in relation to the housing design context in the given site, however in another site or another design task the modules might be organized in a different way by other researchers. Putting it another way, the proposed framework can be adapted to different design contexts by adding, subtracting or modifying objective criteria. Fitness function can be defined in different design areas according to needs. The use of both non-dominated multi-objective genetic algorithms and the use of a dynamic / adaptive modular framework can help build expertise and experience in similar design areas, especially in design offices, in early steps of the design process.

It became a common knowledge that the flexibility of the initial parameters and fitness function in the multi optimization design problems have a crucial impact on the richness of the solution space, in this study particularly 3D mass variations. Results of the case study confirm that multi-objective genetic algorithm have advantage over singular objective genetic algorithm in terms of meeting the required flexibility in design process. SOGA provides solutions focusing on the best generations of one selected hill, on the other hand in MOGA search in global hills goes on. This property of going on searching in global hills led possibility of getting closer to better solution alternatives. Moreover, apart from the simultaneous connection of different design goals in MOGA possibility of changing the weights/impact factor of the design goals interactively creates a better interaction between design alternatives and the designer.

The alternatives obtained from the application of SOGA in case study can be criticized in terms of morphological similarities. The reason for this is the dependence to the initial assumptions, such as the regulation objectives and also Galapagos’ solving process. The maximum floor area and the use of constructible volume are strictly defined in the selected site, but the results would be different in another study. If the architect is expected to produce design alternatives within a limited time on a given site, the proposed framework has the potential to look for 3D image possibilities and give quick feedback to the designers.

Finally, Octopus and Galapagos both have potentials of gathering outputs in the multi optimization design processes with little differences. Octopus can generate in a wide range of alternatives which can be beneficial in form search problems and Galapagos have a strong meet the needs of fitness function faster than Octopus. So in some cases Galapagos could be preferred to get faster solutions and Octopus for variety.

References

variety. Some cases Galapagos could be preferred to get faster solutions and Octopus for Galapagos have a strong meet the needs of fitness function faster than Octopus. So in wide range of alternatives which can be beneficial in form search problems and multi optimization design processes with little differences. Octopus can generate in possibilities and give quick feedback to the designers.

Finally, Octopus and Galapagos both have potentials of gathering outputs in the given site, the proposed framework has the potential to look for 3D image If the architect is expected to produce design alternatives within a limited time on a strictly defined in the selected site, but the results would be different in another study. Criticized in terms of morphological similarities. The reason for this is the dependence between design alternatives and the designer. Weights/impact factor of the design goals interactively creates a better interaction closer to better solution alternatives. Moreover, apart from the simultaneous goes on. This property of going on searching in global hills led possibility of getting generations of one selected hill, on the other hand in MOGA search in global hills advantage over singular objective genetic algorithm in terms of meeting the required results of the case study confirm that multi-objective genetic algorithm have the richness of the solution space, in this study particularly 3D mass variations. Fitness function in the multi optimization design problems have a crucial impact on the evolution of design modules. The modules are selected in relation to the housing design context models, designer-specified goals, domain-specific goals based on building types as modules such as regulations, environmental condition analysis as static, behavioral (MOGA) and its comparison with a single objective genetic algorithm (SOGA) in a

References

CAMBRIA: Interacting with Multiple CAD Alternatives

Siniša Kolarić 1,,* (orcid.org/0000-0002-7638-7446), Halil Erhan 2 (orcid.org/0000-0002-7630-6351), and Robert Woodbury 3 (orcid.org/0000-0002-4543-6647)
1, 2, 3 School of Interactive Arts and Technology Simon Fraser University, Surrey, Canada
1 skolaric@sfu.ca, 2 herhan@sfu.ca, 3 rw@sfu.ca

Abstract. Computer-aided design (CAD) tools aim to assist designers in their professional work, one key aspect of which is devising, evaluating, and choosing among multiple design alternatives. Yet, with few and limited exceptions, current tools handle just a single design model at a time, forcing users to adopt various ad hoc tactics for handling multiple design alternatives. Despite considerable prior work, there are no general, effective strategies for supporting design alternatives. New tools are needed to develop such strategies: to learn how designers’ behavior change with support for multiple alternatives. In this article, we describe CAMBRIA, a multi-state prototype tool we developed for working with multiple 2D parametric CAD models in parallel. We describe the outcomes of an analytical evaluation of CAMBRIA using the Cognitive Dimensions framework.

Keywords: Computer-aided design • CAD • Parametric CAD • Interaction design